Amendments to the Specification:

Please replace the paragraph [0017] with the following amended paragraph:

[0017] An important consideration is the retention time of the DRAM cell. Since all DRAM cells are written sequentially and then read sequentially, the amount of time data written to each cell can exceed the retention time of the DRAM cell. For example given a 120ns clock cycle, 8192 wordlines, a 3 clock write cycle and a 4 clock read cycle, data in the very first DRAM cell written has been held by that cell for about 2.95 milliseconds (120ns x 3 x 8192) before being read and the very last DRAM cell written has been held for about 2.96 milliseconds ((120ns x 3 x 8192) + (8192-1) before being read. Given a typical burn-in retention time specification of about 3 milliseconds, there is no retention time problem. However, if the clock cycle is 160ns, then the times are 3.93 (160ns x 3 x 8192) milliseconds and 3.94 ((120ns x 3 x 8192) + (8192-1) milliseconds respectively there is [[no]] a retention time problem.

Applicants are correcting a typographical error in paragraph [0017] and that this change does not constitute new matter. Applicants point out that since the times of 3.93 milliseconds and 3.94 milliseconds significantly exceed the specification of 3 milliseconds cited in paragraph [0017] that there must, by definition, be a retention time problem.

Please replace the paragraph [0023] with the following amended paragraph:

[0023] After all cells have been written, they are read out. Each read sequence takes two clock cycles. At the rising edge of the first clock a bank activate command is issued and at the rising edge of the second clock a RD/AP command is issued. However, data is not present at the output of the bitline amplifiers until the rising edge of the first clock of the next read sequence.

Thus, the RD CAS latency of the DRAM has been reduced from 2 to one clock cycles. A precharge command is issued on the falling edge of the ~~seconf~~ second clock (after the data has been sensed) so the bitline is in a known state and ready for the next read command. The WR/AP command eliminates the need for a PRE command to be [[is]] issued to bring the bitline to a known state preparatory for the next read command.

Applicants are correcting a typographical error in paragraph [0017].